Summary: This paper evaluates logistic and naïve Bayes classification models applied to 3 binary indicators predicting a binary response variable. Decision threshold determination, model comparison, feature prioritization and management recommendation are the objectives.

Research design: Client **default**, **housing**, **loan** experience, and **response** to term-deposit offer are 4 Bernoulli variables selected from 17 attributes collected from 4521 instances of customer interaction. There are no missing values; thus, **data exploration** consists entirely of (1) **cross tabulation** to inform variable and threshold selection and (2) comprehension of possible **variable combinations**. **Data preparation** includes a train / test split. **Evaluation** compares 3-fold cross validation, output receiver operator characteristic (ROC) curves and other measures for Logistic Regression (LogR) and Bernoulli naïve Bayes (BnB) models.

Data Exploration: (1) 11.6% of clients subscribed to deposits, denoted "positive"; a predictive model labelling all clients "negative" would be thusly be 88.4% accurate, but management cannot assume all customers decline the offer, and so, we need to find a better measure.

```
chi2
default pval = 1.0
```

default	no	yes
response		
no	98.3	1.7
yes	98.3	1.7
All	98.3	1.7

```
chi2
housing pval = 0.04
```

housing	no	yes
response		
no	41.5	58.5
yes	57.8	42.2
All	43.4	56.6

```
chi2
loan pval = 0.2
```

loan	no	yes
response		
no	83.8	16.2
yes	91.7	8.3
All	84.7	15.3

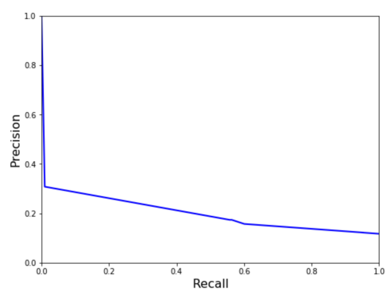
From these 3 **cross tabulations** above, see that clients with **default** experience make up only 1.7% of total and have nearly **identical deposit** take up as those who do **not default** and as the **overall sample**: 88.2% of default and 88.5% of no default clients say no to deposits. There is probably little predictive value here. We would **expect** 52.4% of clients, regardless of deposit response, to have **housing** experience and 13.2% **loan** experience if there were no predictive power for these variables. As can be seen from the cross tab tables **above**, **housing % deviates most from expected percentages and thus carries the lowest p-value, 4%, of the 3 predictors**.

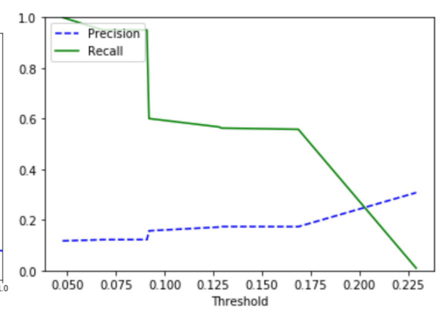
	def	housing	loan	response	LogRprob	BnBprob
1	0	1	1	0	0.047513	0.048195
271	0	1	1	1	0.047513	0.048195
48	1	1	1	0	0.068139	0.067010
2	0	1	0	0	0.090874	0.090731
33	0	1	0	1	0.090874	0.090731
15	0	0	1	0	0.091944	0.093361
115	0	0	1	1	0.091944	0.093361
71	1	1	0	0	0.127798	0.123988
295	1	1	0	1	0.127798	0.123988
403	1	0	1	0	0.129240	0.127446
558	1	0	1	1	0.129240	0.127446
0	0	0	0	0	0.168672	0.168694
13	0	0	0	1	0.168672	0.168694
214	1	0	0	0	0.229235	0.223503
1440	1	0	0	1	0.229235	0.223503

(2) In the **table above** are **8 possible combinations** of the 3 binary predictors. The table contains 15 rows that account for 2 binary response values per predictor combination in the Train set. Three observations: (a) LogR and BnB have similar predicted probabilities per combination and which range 4-23%, (b) There are 7 instances of 2 rows of identical predictor combinations and predicted probabilities where the actual Bernoulli response takes 2 different values suggesting weak predictive power. (c) The brevity in predicted probabilities provdes fewer plot points than needed for continuous ROC and other performance plots.

Python code: Data preparation randomly divides the data set 80% train / 20% test. Although housing appears to have more predictive power, random sampling was chosen over stratification. A confusion matrix was computed but yielded zeros in the Predict True column due to the model's default 50% threshold. The **table below** compares 25 different decision thresholds from which we can see that 10-12% threshold presents **performance maximums**: 63% accuracy, 57% recall, 17% precision and a ROC AUC = 60%. **Any customer whose predicted probability exceeds 10-12% should be regarded as a possible deposit prospect.** We can see

	threshold	accuracy	recall	precision	roc_auc_score
0	1.0	0.116980	1.000000	0.116980	0.500000
1	2.0	0.116980	1.000000	0.116980	0.500000
2	3.0	0.116980	1.000000	0.116980	0.500000
3	4.0	0.116980	1.000000	0.116980	0.500000
4	5.0	0.193308	0.950355	0.121892	0.521685
5	6.0	0.193308	0.950355	0.121892	0.521685
6	7.0	0.194967	0.950355	0.122114	0.522625
7	8.0	0.194967	0.950355	0.122114	0.522625
8	9.0	0.194967	0.950355	0.122114	0.522625
9	10.0	0.629148	0.567376	0.171674	0.602354
10	11.0	0.629148	0.567376	0.171674	0.602354
11	12.0	0.629148	0.567376	0.171674	0.602354
12	13.0	0.636892	0.557920	0.173275	0.602637
13	14.0	0.636892	0.557920	0.173275	0.602637
14	15.0	0.636892	0.557920	0.173275	0.602637
15	16.0	0.636892	0.557920	0.173275	0.602637
16	17.0	0.881637	0.009456	0.307692	0.503319
17	18.0	0.881637	0.009456	0.307692	0.503319
18	19.0	0.881637	0.009456	0.307692	0.503319
19	20.0	0.881637	0.009456	0.307692	0.503319
20	21.0	0.881637	0.009456	0.307692	0.503319
21	22.0	0.881637	0.009456	0.307692	0.503319
22	23.0	0.883020	0.000000	0.000000	0.500000
23	24.0	0.883020	0.000000	0.000000	0.500000
24	25.0	0.883020	0.000000	0.000000	0.500000





these figures in the **Precision Recall plot** (**above LEFT)** (17% precision and 57% recall at the slight dog leg) and in the **Precision Recall plot of Thresholds (above RIGHT**) (where the slope of Recall in green is slightly negative and Precision in blue slightly positive between 10-14% threshold). **Cross validation** employed Scikit-Learns routine which folded the train set into 3 parts and scored the 3 resulting Logistic models for accuracy, resulting in a number = 88.3% in each case, consistent with the results of cross tabulation: over fitting would not be an issue with these indicator variables. A **manually coded cross validation routine** duplicated identical results (employing the default 50% threshold), then lowered the decision threshold to varying levels: within the 10-14% range suggested above, and at the 17 %thresholds implied by the combination of predictors = [0, 0, 0] for which LogR predicted 0.16782 probability in the **table on page 2**. Here are the results:

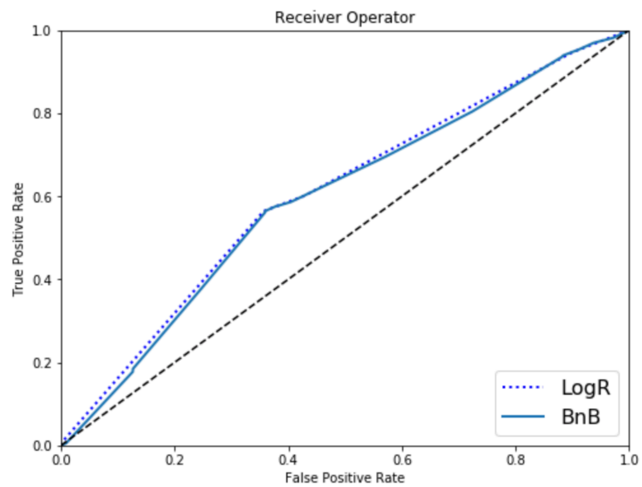
Within the 10-14% threshold range, results were steady:

```
Iteration: 1 Log reg has accuracy: 0.20575221238938052 at 13.0 % threshold
Iteration: 2 Log reg has accuracy: 0.2168141592920354 at 13.0 % threshold
Iteration: 3 Log reg has accuracy: 0.20934734513274336 at 13.0 % threshold
```

Along the 17% threshold, results were more variable as the model struggled to decide:

```
Iteration: 1 Log reg has accuracy: 0.288716814159292 at 17.0 % threshold
Iteration: 2 Log reg has accuracy: 0.2942477876106195 at 17.0 % threshold
Iteration: 3 Log reg has accuracy: 0.20934734513274336 at 17.0 % threshold
```

The same manual cross validation analysis was attempted with naïve Bayes but Scikit-Learn could not clone the fit between tests. As probability profiles for LogR and BnB are similar, I will assume both have similar sensitivities near these pivotal thresholds. I do not assume these results are indicative of underfit, though with more predictors, sensitivity to threshold may be reduced. The **C regularization** hyper parameter was also tested at the 0.01 level where a more stoic model was observed with lower performance measures (not presented here). Lastly, the **ROC curve for both Logistic and Naïve Bayes models** is plotted together with their lines lying nearly on top of one another. as might have been expected based upon the similar probabilities at each combination of binary variables.



Management recommendations. The 11-12% incidence of deposit take-up would not support using nominal predictions from the Logistic or naïve Bayes models, but with adjustment, the Logistic model can prove useful identifying promising customers for deposits. The model from the train set yields Log(odds) on this basis:

0.38*default -0.71*housing -0.69*loan -1.6 intercept (or bias) term.

For example, the client who has a housing loan has a 9% probability of deposit take up based on exp (–0.71*1 – 1.6) odds where probability = odds / (1+odds). That is less than the 16.8% probability = exp(-1.6) / (1+exp(-1.6)) for a client with no services. The **most promising deposit customer** would be one who has just defaulted, with a exp(0.38-1.6)/(1+exp(0.38-1.6) = 22% probability of taking on a new term deposit. In this sense, **housing with the most significant beta has the greatest influence** of the 3 predictors of probability, bearing out conclusions from cross tabulation, though the effect is counterintuitive in each of the betas' cases, where housing and personal loans take up are usually associated with and directly causing deposits of some length or another and defaults the reverse. (It is possible that these interpretations are all backwards and that a 1 for default signifies no default.) In any case, comprehending these characteristic predictors would make sales calls more productive as the 17% precision is an improvement over 11-12% of clients chosen at random. Thus, this and more studies of this kind are very useful for improving the bottom line.